Two Algorithms for Orthogonal Nonnegative Matrix Factorization with Application to Clustering

Filippo Pompili* Nicolas Gillis† P.-A. Absil‡ François Glineur‡§

Abstract

Approximate matrix factorization techniques with both nonnegativity and orthogonality constraints, referred to as orthogonal nonnegative matrix factorization (ONMF), have been recently introduced and shown to work remarkably well for clustering tasks such as document classification. In this paper, we introduce two new methods to solve ONMF. First, we show mathematical equivalence between ONMF and a weighted variant of spherical k-means, from which we derive our first method, a simple EM-like algorithm. Our second method is based on an augmented Lagrangian approach. Standard ONMF algorithms typically enforce nonnegativity for their iterates while trying to achieve orthogonality at the limit (e.g., using a proper penalization term or a suitably chosen search direction). Our method works the opposite way: orthogonality is strictly imposed at each step while nonnegativity is asymptotically obtained, using a quadratic penalty. Finally, we show that the two proposed approaches compare favorably with standard ONMF algorithms on both text and image datasets.

1 Introduction

We are interested in solving nonnegative matrix factorization (NMF) problems with additional orthogonality constraints. Given an m-by-n nonnegative matrix M and a factorization rank k (with $k < \min\{m, n\}$), NMF can be formulated as follows

$$\min_{U \in \mathbb{R}^{m \times k}, V \in \mathbb{R}^{k \times n}} ||M - UV||_F^2 \text{ s.t. } U \geq 0 \text{ and } V \geq 0,$$

i.e., find an m-by-k nonnegative matrix U and an k-by-n nonnegative matrix V such that $M \approx UV$. NMF has become a very popular dimensionality reduction technique, and has been used successfully in many applications, see, e.g., [3, 9] and the references therein. Adding the orthogonality constraint $VV^T = I_k$ leads to another problem called orthogonal nonnegative matrix factorization (ONMF) which has tight connections with data clustering (see Section 2). In particular, empirical evidence suggests that this additional orthogonality constraint can improve clustering performance compared to standard NMF or k-means [5, 19]. Current approaches to solve ONMF problems are typically based on suitable modifications of the algorithms developed for the original NMF problem [5, 8, 19]. They enforce nonnegativity of the iterates at each step, and strive to attain orthogonality at the limit (but never attain exactly orthogonal solutions). In fact, dealing with matrices that are both orthogonal and nonnegative is difficult because the combination of these two properties imposes a sparsity structure that confers a combinatorial aspect to the problem (directly related to clustering indicator matrices, see Section 2), which is not easily handled by standard continuous optimization schemes.

The paper is organized as follows. In Section 2, we analyze the relationship between ONMF and clustering problems and show that it is closely related to spherical k-means. Based on this analysis, we develop an EM-like algorithm which features a rank-one NMF problem at its core. Section 3 introduces another algorithm to perform ONMF using an augmented Lagrangian and a projected gradient scheme, which enforce orthogonality at each step while obtaining nonnegativity at the limit. Finally, in Section 4, we experimentally show that our two new approaches perform competitively with standard ONMF algorithms on text datasets and on different image decomposition problems.

*Department of Electronic and Information Engineering, University of Perugia, Italy; filippo.pompili@diei.unipg.it.

†University of Waterloo, Department of Combinatorics and Optimization, Waterloo, Ontario N2L 3G1, Canada; ngillis@uwaterloo.ca

‡Université catholique de Louvain, ICTEAM Institute, Avenue Georges Lemaitre 4, B-1348 Louvain-la-Neuve, Belgium; pa.absil@uclouvain.be

§Université catholique de Louvain, CORE, Voie du Roman Pays 34, B-1348 Louvain-la-Neuve, Belgium; francois.glineur@uclouvain.be.

2 Equivalence of ONMF with a Weighted Variant of Spherical k-means

In this section, we briefly recall how NMF with an additional constraint is equivalent to a fundamental clustering technique: Euclidean k-means [6, 7]. We then show that relaxing this constraint leads to ONMF, which is therefore not exactly equivalent to k-means but rather to another problem closely related to spherical k-means [2]. Based on this analysis, we propose a new EM-like algorithm to solve ONMF problems.

2.1 Equivalence with Euclidean k-means Let $M = (m_1, \ldots, m_n) \in \mathbb{R}_+^{m \times n}$ be a nonnegative data matrix whose columns represent a set of n points $\{m_j\}_{j=1}^n \in \mathbb{R}_+^m$. Solving the clustering problem means finding a set $\{\pi_i\}_{i=1}^k$ of k disjoint clusters:

$$\pi_i \subseteq \{1, 2, \ldots, n\} \; \forall i, \quad \cup_{1 \leq i \leq k} \pi_i = \{1, 2, \ldots, n\},$$

$$\text{and} \quad \pi_i \cap \pi_j = \emptyset \; \forall i \neq j,$$

such that each cluster π_i contains objects as similar as possible to each other according to some quantitative criterion. When choosing the Euclidean distance, we obtain the k-means problem, which can be formulated as follows [6]:

$$\min_{\{\pi_i\}_{i=1}^k} \sum_{i=1}^{k} \sum_{j \in \pi_i} \|m_j - c_i\|^2,$$

$$\text{where } \; c_i = \frac{\sum_{j \in \pi_i} m_j}{|\pi_i|} \text{ are the cluster centroids.}$$

Equivalently, we can define a binary cluster indicator matrix $B \in \{0, 1\}^{k \times n}$ as follows:

$$B = \{b_{ij}\}_{k \times n} \quad \text{where} \quad b_{ij} = 1 \iff j \in \pi_i.$$

Disjointness of clusters π_i means that rows of B are orthogonal, i.e., BB^T is diagonal. Therefore we can normalize them to obtain an orthogonal matrix $V = \{v_{ij}\}_{k \times n} = (BB^T)^{-\frac{1}{2}} B$ (a weighted cluster indicator matrix) which satisfies the following condition:

(c1) There exists a set of clusters $\{\pi_i\}_{i=1}^k$ such that

$$v_{ij} = \begin{cases} \frac{1}{\sqrt{|\pi_i|}}, & \text{if } j \in \pi_i, \\ 0, & \text{otherwise.} \end{cases}$$

It has been shown in [7] that the NMF problem with matrix V satisfying condition (c1):

$$\min_{U \geq 0, V \geq 0} \|M - UV\|_F^2 \;\; \text{s.t.} \quad V \text{ satisfies (c1)} \;, \tag{2.1}$$

is equivalent to k-means. In fact, since V in problem (2.1) is a normalized indicator matrix which satisfies $v_{ij} = |\pi_i|^{-\frac{1}{2}} \iff j \in \pi_i$, we have

$$\begin{aligned} \|M - UV\|_F^2 &= \sum_{j=1}^{n} ||m_j - \sum_{i=1}^{k} u_i v_{ij}||^2 \\ &= \sum_{i=1}^{k} \sum_{j \in \pi_i} \|m_j - u_i v_{ij}\|^2 \\ &= \sum_{i=1}^{k} \sum_{j \in \pi_i} \|m_j - u_i \frac{1}{\sqrt{|\pi_i|}}\|^2, \end{aligned}$$

which implies that, at optimality, each column u_i of U must correspond (up to a multiplicative factor) to a cluster centroid with $u_i = \sqrt{|\pi_i|}\, c_i = \frac{\sum_{j \in \pi_i} m_j}{\sqrt{|\pi_i|}} \; \forall i = 1, \ldots, k$.

2.2 ONMF and a Weighted Variant of Spherical k-means Let us now define a condition weaker than (c1):

$$VV^T = I_k \quad \text{and} \quad V \geq 0. \tag{c2}$$

It can be easily checked that (c1) $\Rightarrow$ (c2) while (c2) $\not\Rightarrow$ (c1). The difference between conditions (c1) and (c2) is that condition (c2) does not require the rows of V to have their non-zero entries equal to each other. Now, if we only impose the weaker condition (c2) on NMF, we obtain a relaxed version of (2.1) which, by definition, corresponds to orthogonal NMF:

$$\min_{U\geq 0, V\geq 0} \|M - UV\|_F^2 \quad \text{such that} \quad VV^T = I_k. \tag{2.2}$$

In the following, we show the equivalence of problem (2.2) with a particular weighted variant of the spherical k-means problem.

It is well known that given a pair of solution matrices (U, V), one can find solutions with the same objective value $\|M - UV\|_F^2$ by considering the pairs $\left(UD^{-1}, DV\right)$, where D is any diagonal matrix with positive diagonal elements. Using this property, we can put the problem in a form where each column of matrix U has unit ℓ_2-norm. This simply amounts to taking $D = \text{diag}(\|u_1\|, \ldots, \|u_k\|)$. Therefore, (2.2) is equivalent to

$$\begin{aligned} \min_{U\geq 0, V\geq 0} \|M - UV\|_F^2 \ \text{ s.t. } \ & (VV^T)_{ij} = 0 \ \forall i \neq j, \\ & \text{and } \ \|u_i\| = 1 \, \forall i. \end{aligned} \tag{2.3}$$

Now, assuming we are given the set of non-zero entries of V (in the form of a partitioning $\{\pi_i\}_{i=1}^k$ such that $V_{ij} \neq 0 \Leftrightarrow j \in \pi_i$) and the cluster directions u_i (with $||u_i|| = 1$ and $u_i \geq 0$), the optimal V can be computed in closed form. In fact, because rows of V are orthogonal to each other, we have, as before: $\|M - UV\|_F^2 = \sum_{i=1}^k \sum_{j\in\pi_i} \|m_j - u_i v_{ij}\|^2$. For each term $\|m_j - u_i v_{ij}\|^2$, the optimal v_{ij}^* is given by:

$$\begin{aligned} v_{ij}^* &= \operatorname*{argmin}_{x\geq 0} \|m_j - u_i x\|^2 \\ &= \operatorname*{argmin}_{x\geq 0} \left(m_j^T m_j - 2x m_j^T u_i + x^2\right) \\ &= m_j^T u_i, \ 1 \leq i \leq k, j \in \pi_i. \end{aligned} \tag{2.4}$$

Backsubstituting the optimal coefficients (2.4) in (2.3), we can rewrite problem (2.3) as

$$\begin{aligned} &\min_{\{\pi_i, u_i\geq 0, ||u_i||_2=1\}_{i=1}^k} \sum_{i=1}^k \sum_{j\in\pi_i} \|m_j - \left(m_j^T u_i\right) u_i\|^2 \\ &= \sum_{i=1}^k \sum_{j\in\pi_i} \left(m_j^T m_j - 2\left(m_j^T u_i\right)^2 + \left(m_j^T u_i\right)^2\right). \end{aligned}$$

Since the terms $m_j^T m_j$ are constants, we have that problem (2.3) is finally equivalent to

$$\max_{\{\pi_i, u_i\geq 0, ||u_i||_2=1\}_{i=1}^k} \sum_{i=1}^k \sum_{j\in\pi_i} \left(m_j^T u_i\right)^2, \tag{2.5}$$

$$\equiv \max_{\{\pi_i, u_i\geq 0, ||u_i||_2=1\}_{i=1}^k} \sum_{i=1}^k \sum_{j\in\pi_i} \|m_j\|^2 \left(\frac{m_j^T}{\|m_j\|} u_i\right)^2. \tag{2.6}$$

It is insightful to compare formulation (2.6) of ONMF with the spherical k-means problem [2], which is a variant

of k-means where both data points and centroids are constrained to have unit norm:

$$\min_{\{\pi_i,u_i\}_{i=1}^k} \sum_{i=1}^k \sum_{j\in\pi_i} \|\frac{m_j}{||m_j||} - u_i\|^2 \quad \text{s.t.} \quad ||u_i||_2 = 1,$$

$$\equiv \max_{\{\pi_i,u_i\}_{i=1}^k} \sum_{i=1}^k \sum_{j\in\pi_i} \frac{m_j^T}{||m_j||} u_i \quad \text{s.t.} \quad ||u_i||_2 = 1. \tag{2.7}$$

Note that, in both problems (2.6) and (2.7), we are maximizing the cosines of the angles between u_i and the data points from the corresponding cluster. However, we observe that:

- Because of coefficients $||m_i||^2$, problem (2.6) is sensitive to the norm of the data points, as opposed to spherical k-means (2.7) which only depends on their direction;
- Even for normalized data points (i.e., $||m_i|| = 1\ \forall i$), problem (2.6) is similar but not equivalent to spherical k-means (2.7) because it tries to maximize *the sum of squares* of the cosines (instead of their sum).
- Contrarily to problem (2.6), spherical k-means (2.7) does not require nonnegativity of u_i's, although it will clearly hold at optimality when data points m_j are nonnegative.

To summarize, we have the following result:

THEOREM 2.1. *For a nonnegative*[1] *data matrix M, the ONMF problem* (2.2) *is equivalent to the weighted variant of spherical k-means* (2.6).

To illustrate the differences between these different clustering techniques, Figure 1 displays a comparison between k-means, standard spherical k-means and ONMF.

2.3 EM-like Algorithm for ONMF We present here a simple EM-like alternating algorithm designed to solve the ONMF problem (2.2) based on its equivalence with the weighted variant of spherical k-means (2.6). It is very similar to the standard spherical k-means algorithm [2], except for the computation of cluster centroids. Specifically, it starts with an initial set of centroids, either randomly chosen or supplied as initial values. It then alternates between two steps:

1. Given cluster centroids $\{u_i\}_{i=1}^k$, choose $\{\pi_i\}_{i=1}^k$ assigning each point to its closest cluster:

$$\begin{aligned} j \in \pi_i \iff i &= \operatorname*{argmax}_{1\le\ell\le k} \left(m_j^T u_\ell\right)^2 \\ &= \operatorname*{argmax}_{1\le\ell\le k} \left(m_j^T u_\ell\right). \end{aligned}$$

   Notice that this step is exactly equivalent to the one of standard spherical k-means [2].

2. Given the clustering $\{\pi_i\}_{i=1}^k$, compute the new optimal cluster centroids $\{u_i\}_{i=1}^k$ as follows. Define matrix $M_i \in \mathbb{R}^{m\times|\pi_i|}$ as the submatrix of M containing the columns belonging to cluster π_i. We have to solve problem (2.5) with respect to the u_i's:

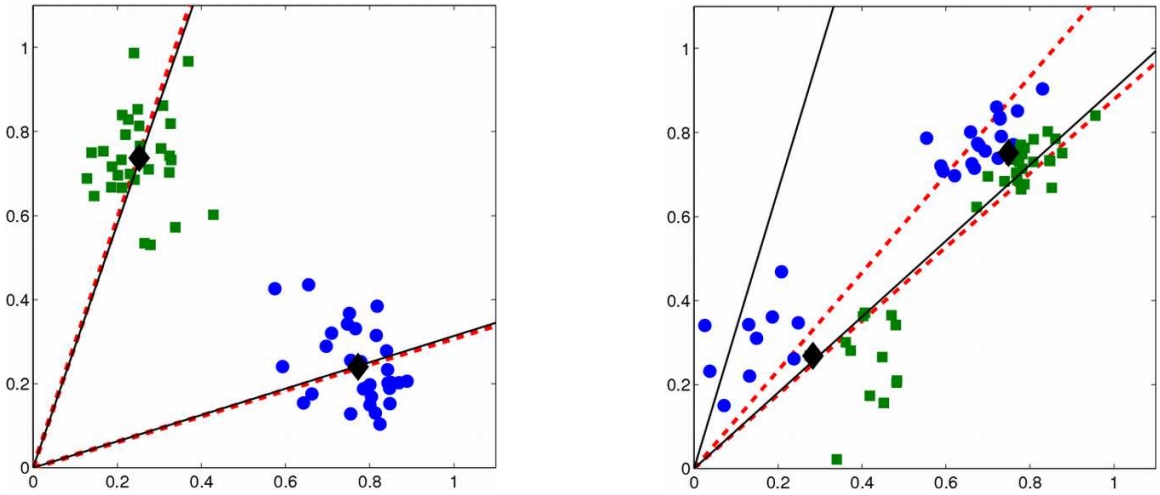
$$\max_{\{u_i\ge 0, ||u_i||=1\}_{i=1}^k} \sum_{i=1}^k \sum_{j\in\pi_i} \left(m_j^T u_i\right)^2 = \sum_{i=1}^k ||M_i^T u_i||_2^2.$$

   There are k independent problems: each u_i must maximize the term $||M_i^T u_i||_2^2$. The optimal solution u_i^* is given by the dominant left singular vector of M_i associated with the largest singular value $\sigma_1(M_i)$ of M_i:

$$u_i^* = \operatorname*{argmax}_{||u||_2=1} ||M_i^T u||_2^2 = \operatorname*{argmax}_{||u||_2=1} u^T M_i M_i^T u,$$

   for which we have $||M_i^T u_i^*||_2 = \sigma_1(M_i) = ||M_i||_2$. Moreover, since $M_i \ge 0$, the Perron-Frobenius theorem guarantees that u_i^* can always be chosen to be nonnegative.

[1] The result also holds for any data matrix M (not necessarily nonnegative) if we remove the nonnegativity constraints on matrix U in the ONMF problem (2.2) and on vectors u_i in problem (2.6).



(a) Well-separated clusters

(b) In-line clusters

Figure 1: Comparison of k-means, standard spherical k-means and ONMF. Diamonds are cluster centroids found by k-means, continuous lines are spherical k-means centroid directions while dashed lines are ONMF centroid directions. Circles and squares are data points as clustered by ONMF. As expected, k-means is not sensitive to the alignment of the clusters as opposed to spherical k-means and ONMF. On the left figure (a), the clusters are well separated and the three techniques perform similarly. On the right figure (b), the directional effect is clearly visible for both ONMF and spherical k-means. However, there is an important difference between the two: ONMF is more sensitive to the data points with larger norm, while spherical k-means treats all the points the same way (including the ones from the lower left cluster with smaller norm but wider angular distribution) and its centroids are therefore further apart from each other.

Algorithm 1, referred to as EM-ONMF, implements this procedure. We will see in the last section that, despite its simplicity, it seems to work well for text clustering tasks.

It is interesting to relate this with the original ONMF problem (2.3): given a partitioning $\{\pi_i\}_{i=1}^k$, let us denote $w_i = (v_{ij})_{j\in\pi_i}$ the subvector containing only the positive entries of the i^{th} row of V. Then,

$$\begin{aligned}\|M - UV\|_F^2 &= \sum_{i=1}^{k}\sum_{j\in\pi_i} \|m_j - u_i v_{ij}\|^2 \\ &= \sum_{i=1}^{k} ||M_i - u_i w_i^T||_F^2,\end{aligned}$$

so that the optimal (u_i,w_i) must be an optimal solution of

$$\min_{||u_i||=1, u_i\geq 0, w_i\geq 0} ||M_i - u_i w_i^T||_F^2. \tag{2.8}$$

Each of these problems looks for the best nonnegative rank-one approximation of a nonnegative matrix (i.e., a rank-one NMF problem). This in turn can be solved by combining the Eckart-Young and Perron-Frobenius theorems: taking the first rank-one factor generated by the singular value decomposition (SVD) (making sure it is nonnegative in case of non-uniqueness) leads to a minimum value for (2.8) equal to $||M_i||_F^2 - \sigma_1^2(M_i)$. Therefore, solving ONMF amounts to finding a partitioning $\{\pi_i\}_{i=1}^k$ such that the sum of squares of the first singular values of submatrices M_i's is maximized, i.e., the ONMF problem (2.2) is equivalent to $\max_{\{\pi_i\}_{i=1}^k} \sum_{i=1}^k \sigma_1^2(M_i)$.

3 Augmented Lagrangian Method for ONMF

In this section, we present an alternative approach to solve ONMF problems. Typically, ONMF algorithms strictly enforce nonnegativity for each iterate while trying to achieve orthogonality at the limit. This can be done using

Algorithm 1: EM-like Algorithm for ONMF (EM-ONMF)

input : Nonnegative data matrix M, and initial centroids $\{u_i\}_{i=1}^k$.
output: Clustering of the points $\{\pi_i\}_{i=1}^k$, with the corresponding centroid directions $\{u_i\}_{i=1}^k$.

while *not converged* **do**
  $\{\pi_i\}_{i=1}^k \leftarrow \emptyset$
  for $j \leftarrow 1$ **to** n **do**
    find $i = \text{argmax}_{1\leq \ell \leq k} \left(m_j^T u_\ell\right)$ and update cluster $\pi_i = \pi_i \cup \{j\}$.
  end
  if $\pi_i = \emptyset$ for some i **then** randomly transfer a point to cluster π_i **end**
  for $i \leftarrow 1$ **to** k **do**
    $u_i \leftarrow$ (any) nonnegative dominant singular vector of the data submatrix $M_i = M(:, \pi_i)$.
  end
end

a proper penalization term [8], a projection matrix formulation [19] or by choosing a suitable search direction [5]. We propose here a method working the opposite way: at each iteration, a (continuous) projected gradient scheme is used to ensure that the V iterates are orthogonal (but not necessarily nonnegative).

Nonnegativity constraints in the ONMF formulation (2.2) will be handled using the following augmented Lagrangian, defined for a matrix of Lagrange multipliers $\Lambda \in \mathbb{R}_+^{k\times n}$ associated to the nonnegativity constraints:

$$L_\rho(U, V, \Lambda) = \frac{1}{2}||M - UV||_F^2 + \langle \Lambda, -V\rangle + \frac{\rho}{2}||\min(V, 0)||_F^2, \tag{3.9}$$

where ρ is the quadratic penalty parameter. Ideally, we would like to solve the Lagrangian dual

$$\max_{\Lambda\geq 0} f(\Lambda) \quad \text{where} \quad f(\Lambda) = \min_{U\geq 0, VV^T = I_k} L_\rho(U, V, \Lambda).$$

Function $f(\Lambda)$ is concave and its maximization (over a convex set) is then a convex problem, see, e.g., [17]. However, evaluating $f(\Lambda)$ exactly (i.e., computing an optimal pair $U^*(\Lambda)$ and $V^*(\Lambda)$) is non-trivial and we propose here instead a simple alternating scheme to update variables U, V and Λ:

1. For V and Λ fixed, the optimal U can be computed by solving a nonnegative least squares problem $U \leftarrow \text{argmin}_{X\in\mathbb{R}_+^{m\times k}} \|M - XV\|_F^2$. We use the efficient active-set method proposed in[2] [13].

2. For U and Λ fixed, we update matrix V by means of a projected gradient scheme. Computing the projection of a matrix $\hat{V}$ onto the feasible set of orthogonal matrices, known as the Stiefel manifold[3], amounts to solving the following problem:
$$\text{Proj}_{St}(\hat{V}) = \underset{X}{\text{argmin}} ||\hat{V} - X||_F^2 \quad \text{s.t. } XX^T = I_k$$
whose optimal solution X^* can be computed in closed form from the unitary factor of a polar decomposition of $\hat{V}$, see, e.g., [12, 1]. Our projected gradient scheme then reads:
$$V \leftarrow \text{Proj}_{St}\Big(V - \beta\nabla_V L_\rho(U, V, \Lambda)\Big),$$
where the step length β is chosen with a backtracking line search similar to that in [14] (step length is increased as long as there is a decrease in the objective function, and decreased otherwise).

3. Finally, the Lagrange multipliers are updated in order to penalize the negative values of V:
$$\Lambda \leftarrow \max\left(0, \Lambda - \alpha V\right),$$

[2]Available at `http://www.cc.gatech.edu/~hpark/`.
[3]The Stiefel manifold is the set of all $n \times k$ orthogonal matrices, i.e., St(k,n) = $\{X \in \mathbb{R}^{n\times k} : X^T X = I_k\}$.

where α is a predefined sequence of step lengths decreasing to zero (e.g., $\alpha = \alpha_0/t$ where t is the iterations count and $\alpha_0 > 0$ is a constant parameter). This can be recognized as an (approximate) subgradient-type scheme [17] (in fact, one can check that V^* is a subgradient of f at Λ when $f(\Lambda) = L_\rho(U^*, V^*, \Lambda)$).

To initialize the algorithm, we set Λ to zero and choose for the columns of V the first k right singular vectors of the data matrix M (which can be obtained with SVD)[4]. Quadratic penalty parameter ρ is initially fixed to a given small value ρ_0 and then increased after each iteration. Alg. 2 implements this procedure, which we refer to as Orthogonal Nonnegatively Penalized Matrix Factorization (ONP-MF). We observed that the term

Algorithm 2: Orthogonal nonnegatively penalized matrix factorization (ONP-MF)

input : A nonnegative data matrix M, the number of clusters k, $\alpha_0 > 0$, $\rho_0 > 0$ and $C > 1$.
output: The centroid matrix U, and the cluster assignment matrix V.

Initialize $\Lambda^{(0)} = 0$, the rows[5] of $V^{(0)}$ with the first k right singular vectors of M, and $\rho = \rho_0$.
for $t = 1, 2, \ldots$ **do**
  Update $U^{(t)}$ with the optimal solution $U^* = \operatorname{argmin}_{U \geq 0} ||M - UV^{(t-1)}||_F^2$.
  Update $V^{(t)}$ with projected gradient and a line search for step $\beta^{(t)}$:

$$V^{(t)} \leftarrow \mathrm{Proj}_{St}\left[V^{(t-1)} - \beta^{(t)} \nabla_V L_\rho\big(U^{(t)}, V^{(t-1)}, \Lambda^{(t-1)}\big)\right].$$

  Update Lagrange multipliers (using an approximate subgradient):

$$\Lambda^{(t)} \leftarrow \max\left(0, \Lambda^{(t-1)} - \frac{\alpha_0}{t} V^{(t)}\right).$$

  Update $\rho \leftarrow C\rho$, where $C > 1$ is a constant parameter.
end

$||\min(V, 0)||_F$ decreases linearly to zero (as augmented Lagrangian methods are expected to, see [15, Th. 17.2]) while $||M - UV||_F$ converges to a fixed value, see Figure 2 for an example on the Hubble dataset (cf. Section 4.2). A rigorous convergence proof is a topic for further research.

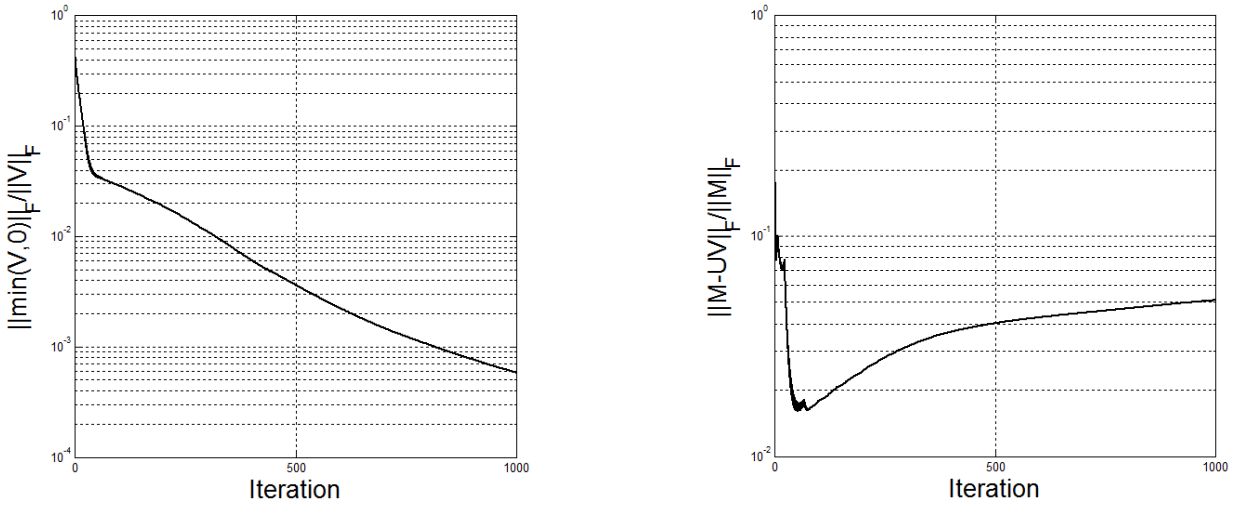


Figure 2: Convergence of Alg. 2 for the Hubble dataset (left: constraint residual, right: approximation error).

[4] To overcome the sign ambiguity of each row of $V^{(0)}$, we flip its sign if the ℓ_2-norm of its negative entries is larger than the ℓ_2-norm of its positive entries.

4 Numerical Experiments

In this section, we report some preliminary numerical experiments showing that ONP-MF (Alg. 2) and EM-ONMF (Alg. 1) perform competitively with two recently proposed methods for ONMF: CHNMF from Choi [5] and PNMF from Yang and Oja [19] (Euclidean variant). It should be noted that because ONP-MF is initialized with SVD, all results are deterministic and obtained with just one execution of the algorithm. However, it could be argued that the comparison is not completely fair since the other ONMF algorithms (namely CHNMF and PNMF) are initialized with randomly generated factors. In order to perform a fairer comparison, we then also initialize CHNMF and PNMF with an SVD-based initialization [4] (SVD cannot be used directly because its factors are not necessarily nonnegative), which will be denoted CH(SVD) and P(SVD) respectively. Finally, we also report the results from two standard EM clustering algorithms, namely k-means and spherical k-means (SKM) (see, e.g., [2]). We will see that EM-ONMF is quite efficient for text clustering tasks (see Section 4.1) while ONP-MF gives very good results for unsupervised image classification tasks (see Sections 4.2 and 4.3).

Parameters for ONP-MF are chosen as follows: $\alpha_0 = 100$, $\rho_0 = 0.01$ and $C = 1.01$ for all datasets. ONMF algorithms are run until a stopping condition is met, or a maximum of 5000 iterations in case of random initializations (for CHNMF and PNMF) and 20000 iterations for the SVD-based initialization (as done in [4]) was reached. The following stopping condition for CHNMF seems to work well in practice[6]:

$$\frac{|\,||M - U^{(t+1)}V^{(t+1)}||_F - ||M - U^{(t)}V^{(t)}||_F|}{||M||_F} < 10^{-7},$$

where t is the iteration count. For PNMF, we use the stopping criterion suggested by its authors[7]:

$$\frac{||V^{(t-1)} - V^{(t)}||_F}{||V^{(t-1)}||_F} < 10^{-5}.$$

For ONP-MF, we check whether the current iterate is 'sufficiently' nonnegative, using

$$\frac{||\min(V, 0)||_F}{||V||_F} < 10^{-3}.$$

All EM-like algorithms, EM-ONMF included, were run until cluster assignment did not change for two consecutive iterations. The initial centroids were randomly selected among the data points. For each experiment, a number of 30 repetitions was executed in random conditions both for ONMF and EM-like algorithms. In the image experiments, we will display the best solution obtained (w.r.t. the error) among the 30 solutions hence obtained. All experiments were run on an Intel® Core™2 Duo CPU @2.40GHz with 6GB of RAM.

4.1 Text clustering We selected nine well-known preprocessed document databases described in [20]. Each dataset is represented by a term-by-document matrix of varying characteristics, see Table 1. As a performance indicator, we use the *accuracy*: given a clustering $\{\pi_i\}_{i=1}^k$ and the true classes $\{L_i\}_{i=1}^k$ of the n elements of the dataset, it is defined by:

$$\text{Accuracy} = \max_{P \in [1,2,\ldots,k]} \frac{1}{n} \left(\sum_{i=1}^{k} |\pi_i \cap L_{P(i)}| \right) \in [0, 1],$$

where $[1, 2, \ldots, k]$ is the set of permutations of $\{1, 2, \ldots, k\}$. We report the average value of the obtained accuracy in Table 2. For more than half of the datasets the average best result was achieved by our algorithms, either EM-ONMF or ONP-MF. Moreover, our algorithms always obtain the best performance among ONMF algorithms. Table 3 reports the average number of iterations of each method, and Table 4 the average computational time. While EM-ONMF is very fast with a low number of iterations (as the other EM-like algorithms), ONP-MF is in general slower than the other ONMF algorithms (especially CHNMF), and typically performs a larger number of iterations to converge. A similar behavior will be observed on the image decomposition tasks (see Table 5 and 6).

[6] It seems that 10^{-7} is a good trade-off: for example, using 10^{-8} instead leads to much larger computational times without significant improvements.

[7] Code available at `http://users.ics.tkk.fi/rozyang/pnmf/index.html`.

Table 1: Text mining datasets [20].

Data	m	n	r	#nonzero
classic	7094	41681	4	223839
ohscal	11162	11465	10	674365
hitech	2301	10080	6	331373
reviews	4069	18483	5	758635
sports	8580	14870	7	1091723
la1	3204	31472	6	484024
la2	3075	31472	6	455383
k1b	2340	21839	6	302992

Table 2: Average accuracy obtained by the different algorithms (in bold, best performance; underlined, second best).

Dataset	k-means	SKM	CHNMF	CH(SVD)	PNMF	P(SVD)	EM-ONMF	ONP-MF
classic	**0.620**	0.577	0.544	0.559	0.536	0.547	<u>0.578</u>	0.538
ohscal	0.281	**0.418**	0.339	0.339	0.342	0.338	<u>0.387</u>	0.340
hitech	0.318	0.484	0.427	0.458	0.414	<u>0.485</u>	**0.494**	0.470
reviews	0.456	**0.678**	0.503	0.494	0.528	0.533	<u>0.655</u>	0.510
sports	0.402	0.466	0.435	0.430	0.491	0.489	<u>0.496</u>	**0.500**
la1	0.352	0.473	0.504	0.444	0.583	<u>0.634</u>	0.503	**0.658**
la2	0.336	0.476	0.446	0.422	0.466	<u>0.508</u>	0.463	**0.528**
k1b	0.707	0.657	0.738	0.606	0.746	<u>0.757</u>	0.735	**0.790**

4.2 Hyperspectral Unmixing A hyperspectral image is a set of images of the same object or scene taken at different wavelengths. Each image is acquired by measuring the reflectance (i.e., the fraction of incident electromagnetic power reflected) of each individual pixel at a given wavelength. The aim is to classify the pixels in different clusters, each representing a different material. We want to cluster the columns of a wavelength-by-pixel reflectance matrix so that each cluster (a set of pixels) corresponds to a particular type of material.

4.2.1 Hubble Telescope We first use a synthetic dataset from [16], see Figure 3 (top row), in clean conditions (i.e., without noise or blur). It represents the Hubble telescope and is made up of 8 different materials, each having a specific spectral signature. Figure 3 displays the clustering obtained by the different algorithms[8] and

[8]For EM-ONMF, k-means and SKM we preprocess the data by discarding pixels from the background (i.e., all columns of the input matrix with zero ℓ_2-norm). Recall that, for each algorithm, we keep the best solution (w.r.t. the error) among the 30 randomly generated initial matrices.

Table 3: Average number of iterations for the different algorithms in the text clustering task.

Dataset	k-means	SKM	CHNMF	CH(SVD)	PNMF	P(SVD)	EM-ONMF	ONP-MF
classic	17	26	254	401	1557	1841	16	2272
ohscal	44	29	354	955	3331	1729	28	2651
hitech	18	21	297	619	2752	2098	20	2476
reviews	20	15	121	149	958	859	13	2581
sports	31	33	462	882	1777	3452	23	2662
la1	20	21	272	668	3197	2550	22	2556
la2	19	20	184	685	2670	3309	19	2541
k1b	13	19	182	39	1816	1846	12	2434

Table 4: Average running time in seconds for the different algorithms in the text clustering task.

Dataset	k-means	SKM	CHNMF	CH(SVD)	PNMF	P(SVD)	EM-ONMF	ONP-MF
classic	1.5	0.3	4	7	31	38	12	216
ohscal	16.0	1.1	28	73	349	171	24	565
hitech	2.4	0.2	7	14	83	60	7	157
reviews	5.8	0.4	5	6	54	48	10	294
sports	15.1	1.5	34	66	184	358	24	478
la1	4.2	0.5	11	27	161	130	19	359
la2	3.8	0.5	7	26	130	161	15	310
k1b	1.8	0.3	5	1	66	67	6	244

Table 5: Average number of iterations for the different algorithms in image decomposition tasks.

Dataset	k-means	SKM	CHNMF	CH(SVD)	PNMF	P(SVD)	EM-ONMF	ONP-MF
Hubble	19	15	1025	423	241	20000	17	2209
Urban	59	66	2209	2645	5000	20000	77	3896
Swimmer	17	20	560	187	164	20000	15	1531

we observe that only ONP-MF is able to successfully recover all eight materials without any mixing. Even with the SVD-based initialization, CHNMF and PFNMF (i.e., CH(SVD) and P(SVD)) are not able to separate all materials properly; ONP-MF is the only algorithm able to perform this task (almost) perfectly.

4.2.2 Urban Dataset The Urban hyperspectral image is taken from HYper-spectral Digital Imagery Collection Experiment (HYDICE) air-borne sensors. It contains 162 clean bands, and 307×307 pixels for each spectral image; it is mainly composed of 6 types of materials: road, dirt, trees, roofs, grass and metal (mostly metallic rooftops) as reported in [11, 10]. The first row of Figure 4 displays a very good clustering obtained using N-FINDR5 [18] plus manual adjustment from [11], along with the clusterings obtained with the different algorithms. The road and dirt are difficult to extract because their spectral signatures are similar (up to a multiplicative factor), and none of the algorithms is able to separate them perfectly. ONP-MF successfully extracts the grass, trees, and roofs and is the only algorithm able to extract the metal (second basis element), while only mixing the road and dirt together. Spherical k-means, CHNMF, P(SVD) (Figure 5) and EM-ONMF also perform relatively well, being able to extract the road (mixed with dirt or metal), trees, grass (as two separate basis elements) and roofs. CH(SVD) and k-means perform relatively poorly: they are not able to separate as many materials properly.

4.3 Image Segmentation: Swimmer Dataset The swimmer image dataset consists of 256 binary images of a body with 4 limbs which can be each in 4 different positions. The goal is to find a part-based decomposition of these images, i.e., isolate the different constitutive parts of the images (the body and the limbs, 17 in total). Moreover, these parts are not overlapping, and therefore no rows of V can share non-zero entries in the same column, and ONMF is the appropriate model. Figure 6 displays the basis elements obtained with the different ONMF algorithms. It can be observed that, in this case, the SVD-based initialization is of no benefit, neither for CHNMF nor for PNMF. All algorithms are able to successfully find the correct parts except PNMF, CH(SVD) and P(SVD).

Table 6: Average running time in seconds for the different algorithms in image decomposition tasks.

Dataset	k-means	SKM	CHNMF	CH(SVD)	PNMF	P(SVD)	EM-ONMF	ONP-MF
Hubble	0.3	0.07	43	6.0	220	445	2.8	106
Urban	42	14	684	267	1582	2839	493	1174
Swimmer	0.1	0.09	5.4	0.2	1.7	28	2	14

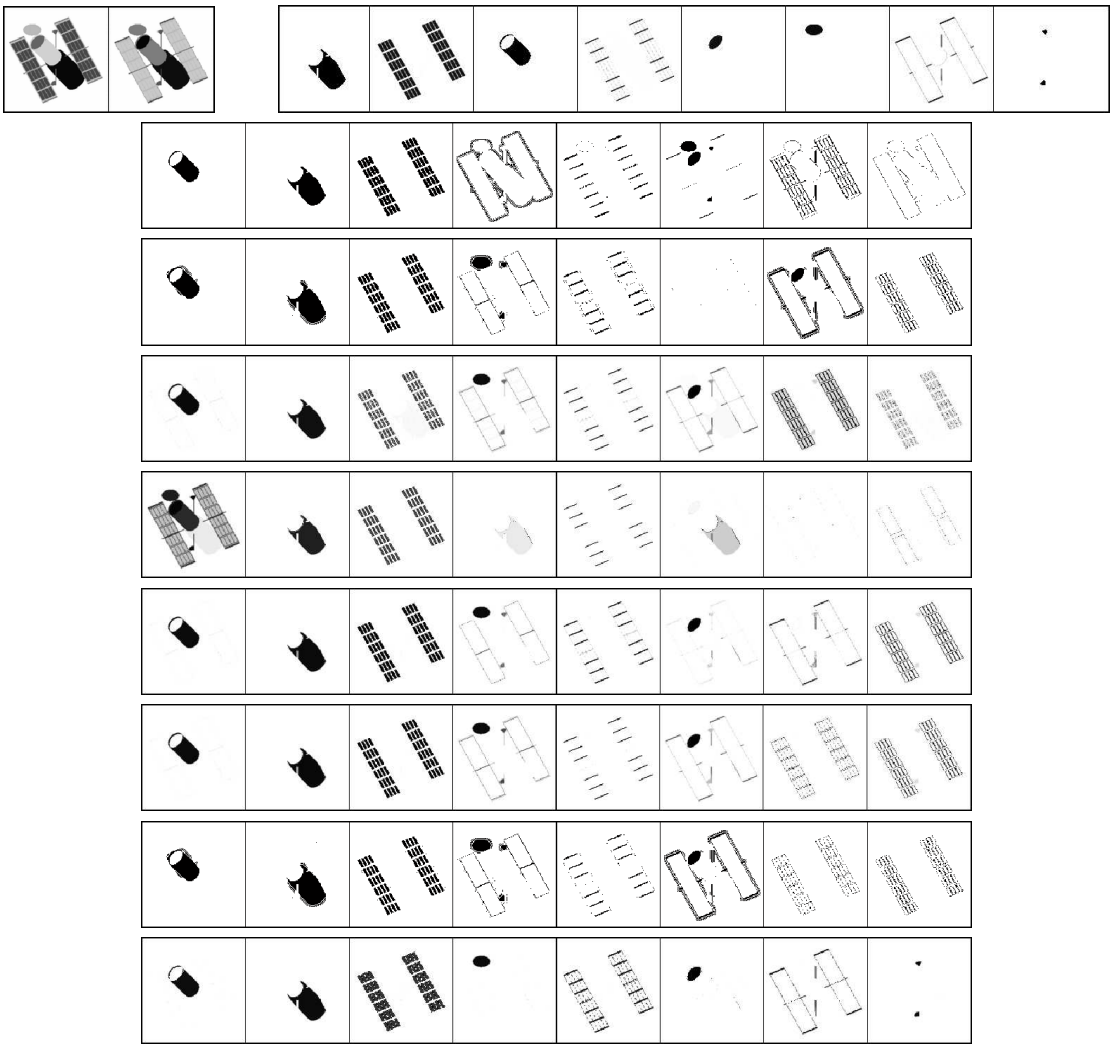


Figure 3: Hubble dataset decomposition. From top to bottom: sample images at different wavelengths along with the true constituent materials; k-means, spherical k-means, CHNMF, CH(SVD), PNMF, P(SVD), EM-ONMF and ONP-MF.

5 Conclusion

In this paper, we have studied the ONMF problem and showed its equivalence with a weighted variant of spherical k-means. This led us to design a new EM-like algorithm for solving ONMF problems (Alg. 1). We have also proposed an alternative approach based on an augmented Lagrangian method imposing orthogonality at each step while relaxing the nonnegativity constraint (Alg. 2). We finally showed on some text and image datasets that these new techniques compare favorably with existing ONMF algorithms. Our ONP-MF algorithm is by far the most robust: it always gave very good results, the best in many cases, using only one initialization. In particular, a single (deterministic) run of ONP-MF worked better in all image experiments than the other algorithms, despite the fact that they were allowed to keep the best solution obtained from 30 different (random) initializations. Further work includes improving its computational efficiency, on which we are currently working using more sophisticated optimization techniques.

References

[1] P.-A. Absil, and J. Malick, *Projection-like Retractions on Matrix Manifolds*, accepted for publication in SIAM Journal on Optimization.

[2] A. Banerjee, I. Dhillon, J. Ghosh, and S. Sra, *Generative Model-based Clustering of Directional Data*, In Pro. of the ninth ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (2003), pp. 19–28.

[3] M.W. Berry, M. Browne, A. Langville, V.P. Pauca, and R.J. Plemmons, *Algorithms and Applications for Approximate Nonnegative Matrix Factorization*, Comp. Stat. and Data Anal. 52 (2007), pp. 155–173.

[4] C. Boutsidis and E. Gallopoulos, *SVD Based Initialization: A Head Start for Nonnegative Matrix Factorization*, Pattern Recognition 41(4) (2008), pp. 1350–1362.

[5] S. Choi, *Algorithms for Orthogonal Nonnegative Matrix Factorization*, In Proc. of the Int. Joint Conf. on Neural Networks (2008), pp. 1828-1832.

[6] I.S. Dhillon, Y. Guan, and B. Kulis, *Weighted Graph Cuts without Eigenvectors: A Multilevel Approach*, IEEE Trans. on Pattern Analysis and Machine Intelligence, 29(11) (2007), pp. 1944–1957.

[7] C. Ding and X. He, *On the Equivalence of Nonnegative Matrix factorization and Spectral Clustering*, In Proc. of the Fifth SIAM Conf. on Data Mining (2005), pp. 606–610.

[8] C. Ding, T. Li, W. Peng, and H. Park, *Orthogonal Nonnegative Matrix Tri-Factorizations for Clustering*, In Proc. of the Twelfth ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (2006), pp. 126–135.

[9] N. Gillis, *Nonnegative Matrix Factorization: Complexity, Algorithms and Applications*, PhD Thesis, Université catholique de Louvain (2011).

[10] N. Gillis, and R.J Plemmons, *Dimensionality Reduction, Classification, and Spectral Mixture Analysis using Nonnegative Underapproximation*, Optical Engineering, 50 (2011), 027001.

[11] Z. Guo, T. Wittman, and S. Osher, *L1 Unmixing and its Application to Hyperspectral Image Enhancement*, Proc. SPIE Conf. on Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XV (2009).

[12] R.A. Horn and C. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, 1990.

[13] J. Kim and H. Park, *Nonnegative Matrix factorization based on Alternating Nonnegativity Constrained Least Squares and Active Set Method*, SIAM J. Matrix Analysis Applications 30(2) (2008), pp. 713–730.

[14] C.-J. Lin, *Projected Gradient Methods for Nonnegative Matrix Factorization*, Neural Computation (19) (2007), MIT press, pp. 2756–2779.

[15] J. Nocedal and S.J. Wright, *Numerical Optimization, Second Edition*, Springer, New York, 2006.

[16] V.P. Pauca, J. Piper, and R.J. Plemmons, *Nonnegative Matrix Factorization for Spectral Data Analysis*, Linear Algebra and its Applications 406 (1) (2006), pp. 29–47.

[17] N.Z. Shor, *Minimization Methods for Non-differentiable Functions*, Springer Series in Computational Mathematics, 1985.

[18] M. Winter, *N-FINDR: An Algorithm for Fast Autonomous Spectral End-member Determination in Hyperspectral Data*, Proc. SPIE Conf. on Imaging Spectrometry V (1999).

[19] Z. Yang and E. Oja, *Linear and Nonlinear Projective Nonnegative Matrix Factorization*, IEEE Trans. on Neural Networks 21 (2010), pp. 734–749.

[20] S. Zhong and J. Ghosh, *Generative Model-based Document Clustering: A Comparative Study*, Knowledge and Information Systems 8(3) (2005), pp. 374–384.

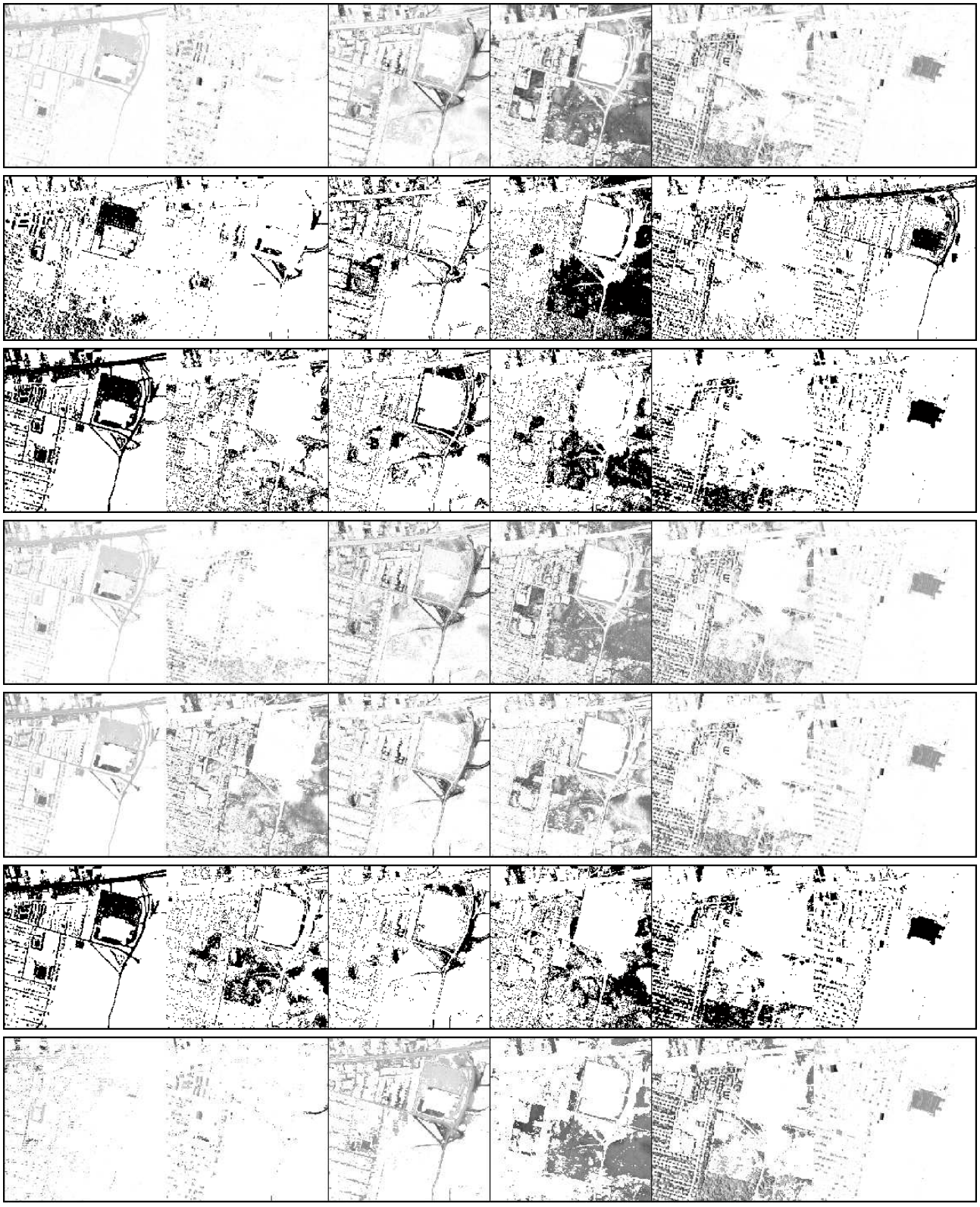


Figure 4: Urban dataset decomposition. From top to bottom: 'true' materials, k-means, spherical k-means, CHNMF, PNMF, EM-ONMF and ONP-MF.

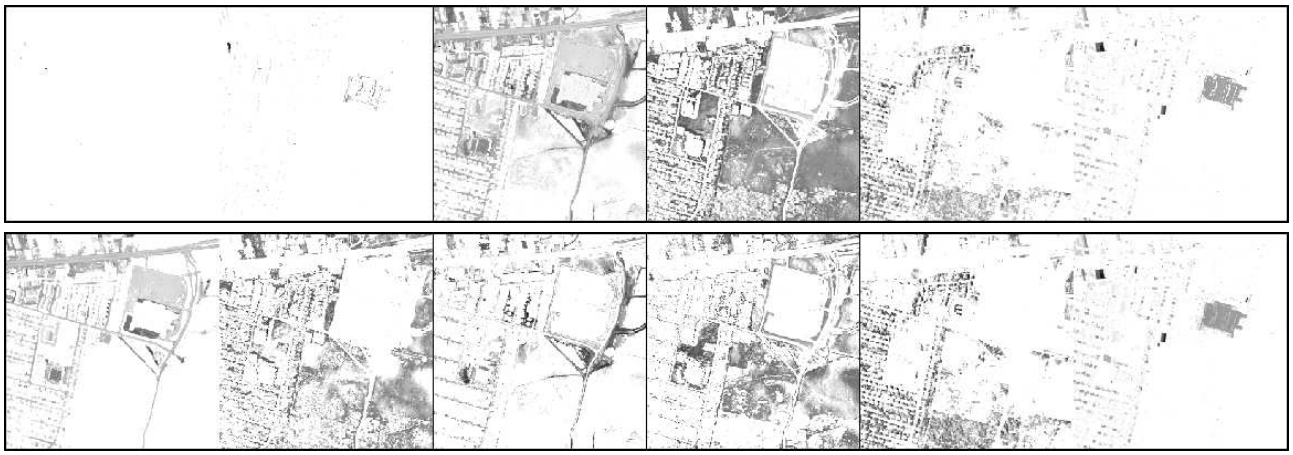


Figure 5: Urban dataset decomposition. From top to bottom: CH(SVD) and P(SVD).

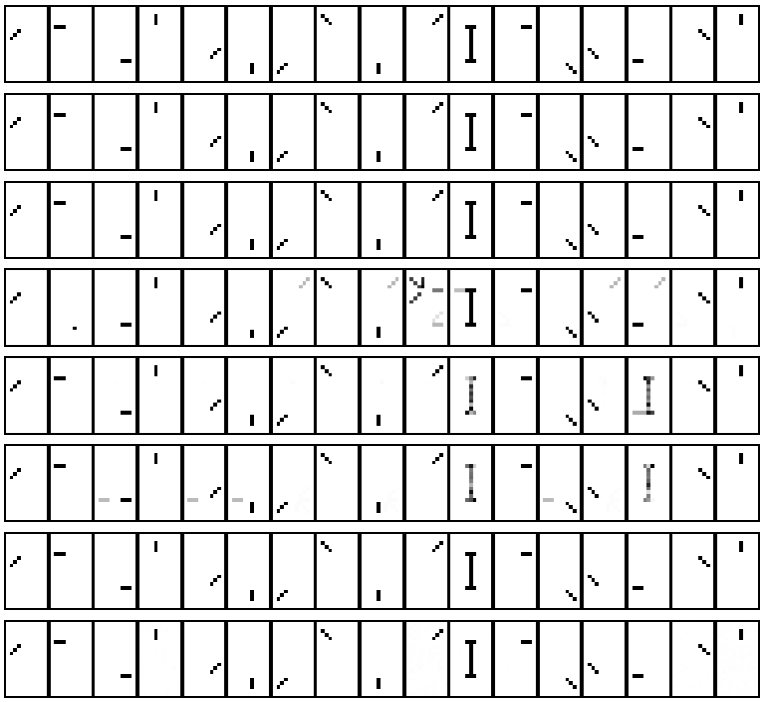


Figure 6: Swimmer dataset decomposition. From top to bottom: k-means, spherical k-means, CHNMF, CH(SVD), PNMF, P(SVD), EM-ONMF and ONP-MF.